

課題名(タイトル):

FX100 を用いたチェックスイート性能評価

利用者氏名:

○南 一生(1,2), 黒田 明義(1,2), 熊畑 清(2,1), 安藤 和人(1,3)

理研における所属研究室名:

- (1) 計算科学研究センター 運用技術部門 チューニング技術ユニット
- (2) 計算科学研究センター フラッグシップ 2020 プロジェクト アプリケーション開発チーム
- (3) 計算科学研究センター 複雑現象統一的解法研究チーム

1. 本課題の研究の背景、目的、関係するプロジェクトとの関係

本課題では、スーパーコンピュータ「京」(以下「京」と記す)並びに富岳スーパーコンピュータでの利活用を促進するために、「京」開発時に作成したチェックスイートやサンプルアプリケーションコードを用いて、FX100 や GPU での性能解析を行い、システム性能の評価、チューニング効果の評価、チェックスイートの問題点の洗い出し、新たなチェックスイートの試作などを行うことを目的に利用を行った。

ここで用いたチェックスイートとは、HPC で利用される計算科学アプリケーションから、主要な計算箇所をループ単位の粒度で切り出したものである。切り出しに用いたアプリケーションは、各分野から、アルゴリズム特性に応じて幅広く選択されている。また用途に応じて、単体チェックスイート、並列チェックスイートに分類出来る。

チェックスイートは、「京」開発当初、システム評価並びにアプリケーションのチューニング指針を得ることを目的に構築された。「京」の運用では、システム(OS、ミドルウェア等)のエンハンス/メンテナンスに伴うアプリケーションの性能劣化を「未然に防止する」「検知する」ことを目的とし、以下の機能整備を行った。

- (1) 複数システム(マシン、コンパイラ)動作
- (2) 計算結果のバリデーションチェック機能
- (3) 性能検証の比較結果レポート機能
- (4) 一括実行機能
- (5) ジョブ多重度管理機能
- (6) 評価カーネル選択機能

現在開発が進められている富岳スーパーコンピュータで

は、最新の比較的アーキテクチャが類似しているシステムで測定された性能をもとに性能評価を行う。このため、現段階で最新のシステムである FX100 並びに GPU を代表とするアーキテクチャを使用した性能評価が重要である。本課題では、現状のチェックスイートを用いた性能評価を実施し、そこから問題点を洗い出し、チェックスイートの再整備を行った。更に実運用で発生した問題の解決に向けたテスト計算を実施した。

2. 具体的な利用内容、計算方法

(1) FrontFlow/blue カーネルの評価

FrontFlow/blue(以降 FFB)は東京大学革新的シミュレーション研究センターで開発された有限要素法による流体シミュレーションコードである。本アプリケーションはものづくりの現場で実用に供されており、今年度も引き続き幾つかの主要演算部分(カーネルと呼称)に着眼して「京」と比しての性能評価ならびに高度化の検討を行った。

(2) ADVENTURE カーネルの評価

ADVENTURE は東京大学で開発されている有限要素法による構造解析アプリケーションである。アプリ開発者から提供された最新版のアプリケーションと、テスト用の入力データ自動生成ツールを用いて、HOKUSAI システムでの性能評価に関わる作業を行った。

(3) OpenFOAM カーネルの評価

産業界で広く利用されている有限体積法による流体解析アプリケーション OpenFOAM についてカーネルの評価とチューニングを行った。OpenFOAM は様々な機能を持つが、本年度は動的メッシュ変形を扱える pimpleDymFOAM

プログラムについてホットカーネルの調査と行いチューニング検討を行った。

(4) 非圧縮性 CUBE のカーネル性能評価・チューニング

計算科学研究センター複雑現象統一的解法研究チームで開発している複雑現象統合シミュレーションフレームワーク CUBE(非圧縮性流体コード版)について、HOKUSAI(GreatWave, BigWaterfall)上でカーネル性能評価及びチューニングを実施した。

(5) ニューラルネットワークを使用した CFD 流れ場の再構成

流れ場に対するPOD(固有値直交分解)をニューラルネットワークで行った論文(Murata et al., 2019)について、CUBEの結果を利用して再現実験を行った。

(6) Chainer の基礎性能評価

Chainer は Preferred Networks 社が開発している Deep Learning による機械学習のフレームワークである。今年度は、「京」や富岳スーパーコンピュータでの Deep Learning の利活用促進に向け、FX100 上での Chainer のフレームワークの基礎性能の評価を実施した。

(7) GENESIS の性能比較

GENESIS は、計算科学研究センター粒子系生物物理研究チームが開発した超並列分子動力学ソフトウェアである。様々な計算環境で高機能かつ高性能な分子動力学計算が可能である。今年度は、CPU として ThunderX2 を搭載した Apollo70 に、GPU として Volta2 枚を付加したマシンでの性能評価の一環として、FX100 での性能を比較のため採取した。

3. 結果

(1) FrontFlow/blue カーネルの評価

東京大学で開発されている有限要素法による流体解析アプリケーション FrontFlow/blue について、本年度は以下の作業を行った。①チューニング統合。②顕著に遅くなっている部分の改善。①は、これまでカーネルを単独で実行できるように切り出した環境で行ったチューニングを、アプリケーション本体への組み込む作業である。組み込みの際して一部データ構造が変更され、更にメインループ実行前に作成しておくデータなどが必要となったため、カーネルのチ

ューニング済み版への入れ替えとともに、アプリケーションの微修正を行った。これによりチューニング前は 300 秒かかっていたメインループ実行時間が、100 秒ほどへと短縮できた。ただし、入力データが異なるため推定値である。②はアプリケーション開発者からの依頼で、メインループ実行前に 1 回だけ実行される準備部分の計算に異常に時間がかかるといった問題の解決である。この部分は形状関数の要素での積分値を前もって計算しておく部分であるが、特定の条件に合致した際に goto 分によりループから抜ける構造となっており、コンパイラによる自動スレッド並列化が想定した位置では働かず、意図しない部分で働いていた。このため、実行時間の 90%以上がスレッド間のバリア同期待ちで消費されていた。goto 文を除去したが、ループ内で用いている配列の参照・更新の依存関係が複雑であったため、意図している部分へのコンパイラによる自動並列化は作用しなかった。最終的にスレッド並列を OpenMP で行うこととし、OpenMP の private/shared 指示文を細かく挿入することで意図したスレッド並列を実現した。結果 1500 秒かかっていた処理を 10 秒以内まで短縮できた。メモリスループの値から考えて、さらに改善出来る可能性はあったが、プログラム実行中に繰り返し実行される部分ではないため、これ以上のチューニングは行わなかった。

(2) ADVENTURE カーネルの評価

東京大学で開発されている有限要素法による構造解析アプリケーション ADVENTURE の性能評価に関わる作業を行った。アプリ開発者から提供された最新版アプリケーションと、テスト用の入力データ自動生成ツールについて、x86 環境ならびに FX100 の環境を用いて、コンパイル・テスト実行の完走を確認した。今後タイマールーチンの挿入ならびにプロファイラを用いて、主要演算部分と通信部分についての調査を行う予定である。

(3) OpenFOAM カーネルの評価

産業界で広く利用されている有限体積法による流体解析アプリケーション OpenFOAM について、本年度は動的メッシュ変形を扱える pimpleDymFOAM プログラムについてホットカーネルの調査と行い、チューニング検討を行った。調査の結果係数行列が疎行列である Gauss-Seidel 法演算がホットカーネルであった。カーネルを切り出して調査した結果、最内ループの回転数がソース上可変として実装され、かつ実行時の回転数が 8 以下と非常に小さいことからソフト

ウェアパイプラインが働いていないと分かった。これは行列が CRS 形式で格納されているためであり、CRS 形式では不可避の問題である。そこで行列格納形式を ELL 形式へと変換し、最内ループの回転数を 8 回転に固定することで、ソフトウェアパイプラインが適応され、実行時間が 1/5 程度に短縮した。

(4) 非圧縮性 CUBE のカーネル性能評価・チューニング

最もコストが高い(全体の 91%を占める)マルチグリッド法によるポアソン方程式のソルバにおいて、最内ループで繰り返し参照する配列を対象とし、下記の 3 つのチューニングを適用した。

[1] 構造体メンバーとして参照していた配列を、ポインタを使用して直接参照する形とすることにより、アドレス計算のための整数演算を削減した。

[2] 配列の次元入れ替えを行うことにより、データアクセスの連続性を改善した

[3] Red-Black ordering で赤と黒に色づけされた配列要素を、それぞれ別の配列に格納する形に変更することにより、データアクセスの連続性を改善した。

これらのチューニングの結果、上記カーネル部分について、GreatWave(FX100, 1CMG)での性能は 12.57 倍となり、また BigWaterfall(Intel Xeon Gold 6148, 1CPU)での性能は 4.5 倍となった。

(5) ニューラルネットワークを使用した CFD 流れ場の再構成

GreatWave 上で 4 つの GPU を使用し、流れ場のスナップショット 10,000 サンプルについてニューラルネットワークの学習を行なった。その結果、論文の記載通り、ニューラルネットワークによって流れ場を 2 つのモードへ分解でき、それらを合成することによって元の流れ場が再構成できることを確認した。

(6) Chainer の基礎性能評価

Deep Learning 計算の計算には GPU が一般に使用される。Chainer は汎用性を重視し、すべての計算が Python で記述され、GPU での計算の以外にも Intel では MKL DNN を使い、その他の CPU では numpy を用いて同等の計算が可能である。我々は、富岳スーパーコンピュータの共用開始時の Deep Learning 利活用促進を目指し、2018 年度は Chainer の性能を分析し、「京」での構築テスト並びに高速

化を手がけた。

2019 年度は、「京」で用いた高速化技法を用いて HOKUSAI FX100 での性能測定を行った。Chainer を FX100 上で手書き数字認識の MNIST サンプル問題を実行したところ、49,166[s]の時間を要した。これは「京」でのチューニングによる実行時間 10,312[s]→283.4[s]と比べて一層低速であるという結果であった。

FX100 でのコスト分布を調べたところ、実行時間のうち 35,089[s]は numpy.dot 計算であり、「京」でのチューニング後の実行時間内訳 131.9[s]と比べて 266 倍遅いことが分かった。numpy.dot は、内部で SGEMM を呼び出している。「京」では、Python を富士通コンパイラで構築し、numpy は内部で SSL2 を呼び出すことで行列積計算を高速化していた。FX100 では、Python を gcc で構築しており、numpy は内蔵コード lapack_lite を呼び出していた。このため、高速な SGEMM ライブラリが利用できず、性能に影響していたと結論づけた。

この他、「京」で行ってきた Adam optimizer の sqrt 計算を Fortran 処理化により、逆数近似展開並びに非正規化数の処理低減を行い、また Python 制御部の threads 非並列部を MPI プロセス並列化することにより、最終的に 47.8[s]、1,029 倍、効率で 13.5%を達成することが確認できた。

この他 ImageNet 相当の画像認識の学習問題のテストも行い、FX100 では「京」同様、Pillow module の導入に問題があることが分かったが、画像 crop などの処理を行わない範囲で ImageNet 相当の学習も可能であることを確認できた。

(7) GENESIS の性能比較

R-CCS で試験的に利用可能になった Volta を 2 枚を搭載する ThunderX2 マシンでのアプリケーション性能評価の一環として、比較のため FX100 での GENESIS の性能を測定した。結果は、198.8[s]と、同じ条件で測定した ThunderX2 の 63.2[s]と比較して 3.1 倍遅いという結果を得た。これにより Volta を搭載した ThunderX2 マシン並びに富岳スーパーコンピュータでの評価結果と比較しえるデータを得た。

4. まとめ

「京」で構築されたチェックスイートやテストコード、テストサンプルを用いて、HOKUSAI GWMPCC や ACSG での評価を

行った。これらの解析を詳細に進めることで、カーネルの更なるチューニングの知見が得られた。FrontFlow/blue や ADVENTURE、OpenFOAM、CUBE のカーネルでの結果はその代表である。また Deep Learning の利活用を目指した調査により、「京」や富岳スーパーコンピュータでも先行している GPU などに対して、十分対抗できる性能を達成でき、実用的な計算の可能性を見込むことができることが分かった。

5. 今後の計画・展望

今回の課題実施によって得られた知見をもとに、更なる詳細の解析を行うとともに、より汎用かつ有用なチェックスイートの整備に力を入れ、京、FX100 並びに富岳スーパーコンピュータの利活用促進に邁進する予定である。

6. 利用がなかった場合の理由

該当なし

2019年度 利用研究成果リスト

【雑誌に受理された論文】

なし

【会議の予稿集】

なし

【口頭発表】

なし

【ポスター発表】

なし

【その他(著書、プレスリリースなど)】

なし